<table>
<tr><td colspan="4">Name of Student: Pushkar Sane</td></tr>
<tr><td colspan="2">Roll Number: 45</td><td colspan="2">Lab Assignment Number: 1</td></tr>
<tr><td colspan="4">Title of Lab Assignment:
To write, test, and debug Basic Python programs.</td></tr>
<tr><td colspan="2">DOP:</td><td colspan="2">DOS:</td></tr>
<tr><td>CO Mapped:
CO1</td><td>PO Mapped:
PO3, PO5, PSO1, PSO2</td><td>Faculty Signature:</td><td>Marks:</td></tr>
</table>

Aim: To write, test, and debug Basic Python programs.

Description: Below are descriptions of basic python modules, functions, etc.

1) **print() Function:**

   For displaying the output to the user on the console, we use the print() function to print the specified message to the screen, or other standard output device.

   Example:

   print(“Hello World”)

2) **Arithmetic operators:**

   Python Arithmetic Operators are used to perform mathematical operations like addition, subtraction, multiplication, and division. The different arithmetic operators are + (Addition), - (Subtraction), * (Multiplication), / (Division), % (Modulus), ** (Exponent).

3) **Input Function:**

   For taking the input from the user, we use the built-in function input() to take the input. Since input() returns a string, we convert the string into numbers using the int() or float() function.

   Example:

   input("Enter your namer: ")

   int(input("Enter your age: "))

4) **Bitwise Operators:**

   In Python, bitwise operators are used to perform bitwise calculations on integers. The integers are first converted into binary and then operations are performed on each bit or corresponding pair of bits, hence the name bitwise operators. The result is then returned in decimal format. The different bitwise operators in python are &(Bitwise AND), | (Bitwise OR), ~ (Bitwise NOT), ^ (Bitwise XOR), << (Bitwise Left Shift), >> (Bitwise Right Shift).

5) **Math Module**

   Python has a built-in module that you can use for mathematical tasks. The math module has a set of methods and constants. For importing a module we use the ‘import’ keyword.

   Example:

   math.factorial(9)

math.sqrt(16)

6) pow() function:

Python pow() function returns the result of the first parameter raised to the power of the second parameter.

Example:

pow(3,2)

print(pow(3,2))

7) Random module:

Python has a built-in module that you can use to make random numbers. The random module has various functions such as randInt, choice, shuffle.

Example:

random.randint(3, 9)

random.choice(x)

random.shuffle(mylist)

8) Calendar module:

Python defines an inbuilt module calendar that handles operations related to the calendar. The calendar module allows us to output calendars like the program and provides additional useful functions related to the calendar. It has various functions such as month, calendar, weekday, etc.

Example:

calendar.month(2024, 3)

calendar.calendar(2018)

9) replace() method:

The replace() method replaces a specified phrase with another specified phrase. If nothing is specified in replace method then all occurrences of specified phrase will be replaced.

Example:

x = txt.replace("one", "three")

10) bin() function:

The bin() is an in-built function in Python that takes in integer x and returns the binary representation of x in a string format.If x is not an integer, then the _index()_ method needs to be implemented to obtain an integer as the return value instead of as a “TypeError” exception.

Example:

```
bin(15)
bin(0xf)
bin(0o17)
```

1) **Add three numbers.**

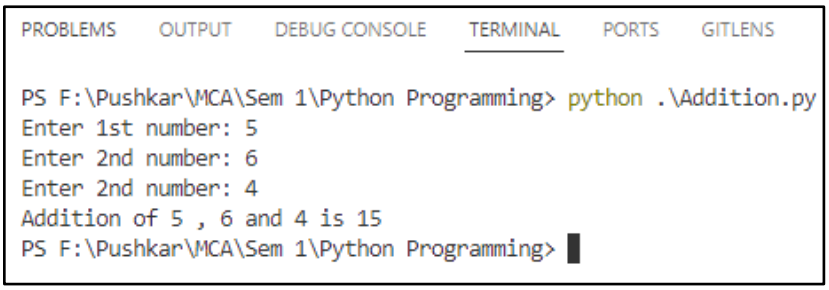
Code:

```
number1 = int(input("Enter 1st number: "))
number2 = int(input("Enter 2nd number: "))
number3 = int(input("Enter 2nd number: "))
print("Addition of" , number1 , ",", number2, "and", number3, "is", (number1 + number2 + number3))
```

Conclusion:

The program for addition of three numbers was executed successfully.

Output:



2) **To swap two numbers using the third variable and without using the third variable.**

Code:

```
# Without using third variable
a = 2
b = 3
a, b = b, a
print("a = ", a, "b = ", b)
# Using third variable
a = 2
b = 3
temp = a
a = b
b = temp
print("a = ", a, " and ", "b = ", b)
```

Conclusion:

We swapped the values without using the third variable in the first program and using the third variable in the second program.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\NumberSwap.py
Without using third variable
a =  3  and  b =  2
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\NumberSwap.py
Using third variable
a =  3  and  b =  2
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

3) Calculate area of triangle.

Code:

```
x = 0.5
base = int(input("Enter the base of triangle: "))
height = int(input("Enter the height of triangle: "))
print("Area of triangle is ", (x * base * height))
```

Conclusion:

The area of the triangle is calculated on the input from the user.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\Triangle.py
Enter the base of triangle: 5
Enter the height of triangle: 4
Area of triangle is  10.0
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

4) To solve Quadratic equations.

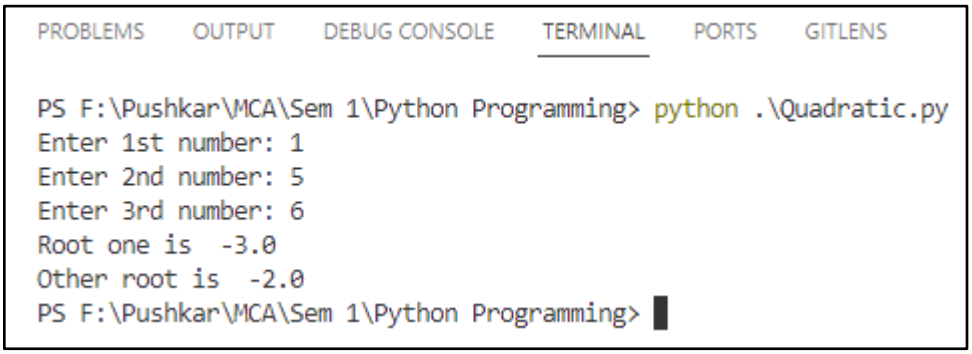
Code:

```
a = int(input("Enter 1st number: "))
b = int(input("Enter 2nd number: "))
c = int(input("Enter 3rd number: "))
# Calculating determinant
x = (b ** 2) - (4 * a * c)
y = x ** 0.5
# Finding roots
if x < 0:
  print("The roots are imaginary.")
else:
  ans1 = (-b - y) / (2 * a)
  ans2 = (-b + y) / (2 * a)
  print("Root one is ", round(ans1, 2))
  print("Other root is ", round(ans2, 2))
```

Conclusion:

The roots of quadratic equations are calculated based on the user's input.

Output:



5) To use Bitwise operators.

Code:

```
a = int(input("Enter 1st number: "))
b = int(input("Enter 2nd number: "))
print("And Operator: ", a & b)
print("Or Operator: ", a | b)
print("Not Operator: ", ~a)
print("XOr Operator: ", (a ^ b))
print("Left shift", a << b)
print("Right shift", a >> b)
```

Conclusion:

We have demonstrated the use of various bitwise operators.

Output:

```
PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\bitwiseop.py
Enter 1st number: 2
Enter 2nd number: 3
And Operator: 2
Or Operator: 3
Not Operator: -3
XOr Operator: 1
Left shift 16
Right shift 0
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

6) To compute compound interest given all the required values.

Code:

```
# Taking values from user
principal = int(input("Enter principal amount: "))
interest = int(input("Enter interest rate: "))
time = int(input("Enter the duration: "))
# Calculating interest
amount = principal * pow(1 + interest / 100, time)
CI = amount - principal
print("The compound interest is ", CI)
```

Conclusion:

We have demonstrated the calculation of compound interest with the given values.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\cinterest.py
Enter principal amount: 100000
Enter interest rate: 7
Enter the duration: 2
The compound interest is 14490.0
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

7) To generate a random number between 0 and 100.

Code:

```
import random
print(random.randint(0, 100))
```

Conclusion:

We have demonstrated generating random number between 0 and 100 with the help of random module.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\RandNumber.py
60
PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\RandNumber.py
21
PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\RandNumber.py
84
PS F:\Pushkar\MCA\Sem 1\Python Programming>
```

8) To display the calendar for January 2024.

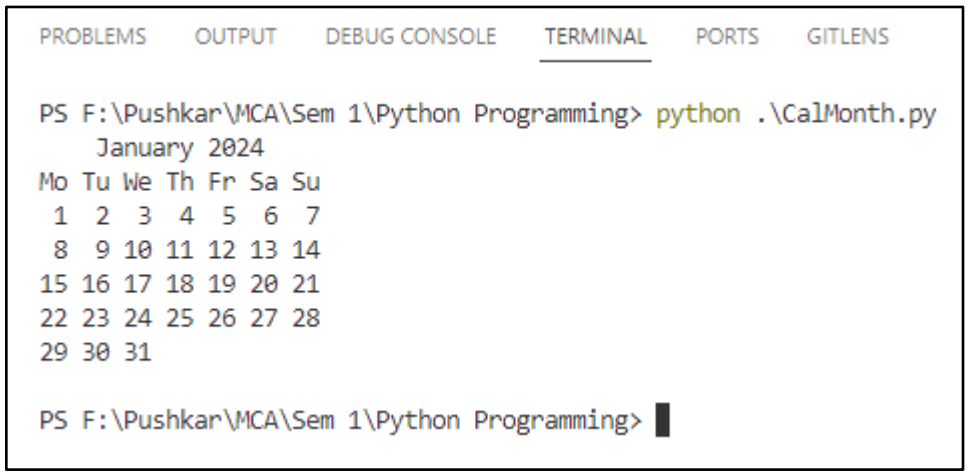
Code:

```
import calendar
print(calendar.month(2024, 1))
```

Conclusion:

We have demonstrated displaying the current calendar year by using the calendar module.

Output:



9) To add two binary numbers.

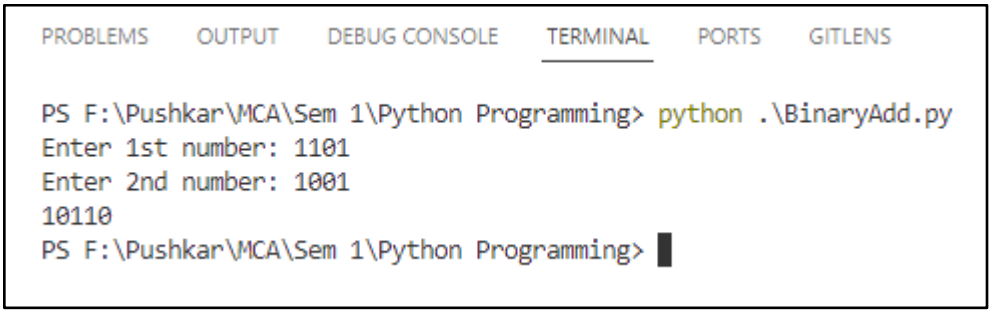
Code:

```
a = input("Enter a binary number: ")
b = input("Enter another binary number: ")
print(bin(int(a, 2) + (int(b, 2))).replace("0b", " "))
```

Conclusion:

We have demonstrated the calculation of adding two binary numbers.

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
GITLENS
PS F:\Pushkar\MCA\Sem 1\Python Programming> python .\BinaryAdd.py
Enter 1st number: 1101
Enter 2nd number: 1001
10110
PS F:\Pushkar\MCA\Sem 1\Python Programming>